

Design of 3D Laser Scanner and Calibration Objects

Undergraduate Honor Thesis

Undergraduate Program in Department of Mechanical Engineering

The Ohio State University

By

Ruiqi Hu

2016

Undergraduate Program in Department of Mechanical Engineering

The Ohio State University

Thesis Committee:

Dr. Sandra Metzler, advisor

Dr. Blaine Lilly

Abstract

3D scanning is commonly used for rapid prototyping, which is a process of converting real world objects into digital models. Most commercial laser scanners in the market are expensive and the scanning process is relatively slow. This research explores the design of an inexpensive 3D laser scanner with medium accuracy for daily use. The purpose of this research is to evaluate the technical performance capability of a DIY laser scanner and compare it to the capability of a commercial scanner. As part of this analysis, a set of standard calibration objects will be developed for a range of volume sizes. The first stage of this research is to construct a portable, inexpensive 3D laser scanner. The scanning system consists of a camera, four line lasers and a turntable. A stepper motor actuates the turntable and an Arduino is utilized in this design as a controller. The Arduino communicates with an open source scanning software program via COM ports. The scanning software is used to perform the image processing and save the scanned image as files which can be utilized for 3D printing by a standard fused deposition modeling device. The second phase of this research aims to improve the accuracy of the scanner by modifying the design. Also, tolerance analysis is also conducted during the design procedure. Finally, in order to evaluate and compare the scanning performance of different scanners, several calibration objects were designed. The result of this study can provide a simple standard method for evaluating the scanning capability of a 3D scanning system, as well as an improved design for an inexpensive 3D laser scanner.

Acknowledgement

First and foremost, it is my greatest honor to have this research opportunity and receive supports from my advisor, Dr. Sandra Metzler. I started from a blank page about the design theorems behind this research. Dr. Metzler has always been patient and knowledgeable, mentoring and guiding me through this research, in which I have a deeper and more systematic insight of Mechanical and Mechatronics Design Application. This research should be considered at least as a synthesis of 2 technical courses in design discipline. Most important, I was able to learn the research and design strategies from Dr. Sandra Metzler. Her advice contributed a lot in achieving my research goal.

Additionally, I want to thank the supervisors in MAE machine shop, Chad Bivens and Kevin Wolf. I could never finish this undergraduate honor research study without their help and advice from manufacturing perspective. They passed a lot of valuable experience to me so I could have a better understanding about how the manufacturing process can affect the design.

Finally, I want to thank College of Engineering in Ohio State University for providing me with scholarships to work on this research project. This scholarship encouraged me a lot when I was facing challenges and difficulties in this research. It is a strong support for both financial aid and problem solving.

Table of Contents

Abstract.....	ii
Acknowledgement.....	iii
Table of Figures.....	v
List of Tables.....	vi
Chapter 1: Introduction	1
1.1 Background	1
1.2 Working Concept of 3D Laser Scanners	2
1.3 Purpose of the Research.....	3
1.4 Significance of the Research.....	3
1.5 Outline of Thesis.....	4
Chapter 2: Design of DIY 3D Scanner	5
2.1 Setting of the DIY Laser Scanner	5
2.2 Control System of the Design.....	6
2.3 Software and Interface	8
Chapter 3: Optimization and Redesign.....	11
3.1 Alignment Part	11
3.2 Tolerance Analysis of the Center Position.....	15
3.3 Other Redesigns of Components	17
Chapter 4: Calibration Objects	18
4.1 Calibration Grid	18
4.2 Motion Capture	19
4.3 Design of Calibration Objects	20
Chapter 5: Result and Validation.....	23
5.1 Scanning Capability of ROMER Absolute Arm	23
5.2 Scanning Capability of Structure Sensor	32
5.3 Scanning Capability of the DIY 3D Laser Scanner	34
Chapter 6: Conclusion and Future Work.....	36
References	38
Appendices	
Appendix A: Pseudocode Instruction to Test the Arduino Sketch	41
Appendix B: Arduino Sketch to Control the DIY 3D Laser Scanner	42
Appendix C: Configuration Header File with Variable Name Defined.....	49
Appendix D: Specification Sheet of Motor 28BYJ-48	51

Table of Figures

Figure 1: Components in 3D Scanning System (A), Integrated 3D Scanner System (B) [3]	1
Figure 2: Sketch of 3D Laser Scanning System.....	2
Figure 3: Top View of the DIY Scanner Physical Setting.....	6
Figure 4: Circuit Sketch of the Control System [5].....	7
Figure 5: Main Interface of Sardauscan [5].....	8
Figure 6: Manual Calibration Interface of Sardauscan [5].....	9
Figure 7: Original Calibration Part	11
Figure 8: Alignment Part	12
Figure 9: Bottom of Alignment Part.....	13
Figure 10: The Alignment Part and Turn Table Assembly	14
Figure 11: Scheme of the tolerance analysis	15
Figure 12: Downsized Laser Holders (A) and Coupler (B).....	17
Figure 13: Commercial Calibration Grid	18
Figure 14: Motion Capture [7].....	19
Figure 15: Calibration Wand [8].....	19
Figure 16: Conceptual Design of Calibration Object 1 (A) and (B)	20
Figure 17: Final Design of Calibration Object 1 (A) and (B)	21
Figure 18: 3D printed Calibration Object 1 (A) and 2 (B)	22
Figure 19: Point Clouds of Calibration Object 1 (A) and 2 (B).....	23
Figure 20: Scan Results of Calibration Object 1 (A) and 2 (B) in SolidWorks	24
Figure 21: Setup of Physical Measurement of Angles on Rotary Table.....	26
Figure 22: The Structure Sensor [9].....	32
Figure 23: Scan of Calibration Object 1 (A) and 2 (B).....	32
Figure 24: Scan of Calibration Object 1 (A) and 2 (B).....	34
Figure 25: Scan of Aluminum box.....	35

List of Tables

Table 1: Measurement of Step Height of Calibration Object 1	25
Table 2: Measurement of Angular positions on Step 1 of Calibration Object 1	27
Table 3: Measurement of Angular positions on Step 3 of Calibration Object 1	28
Table 4: Measurement of Angular positions on Step 5 of Calibration Object 1	29
Table 5: Measurement of Step Height of Calibration Object 2	30

Chapter 1: Introduction

1.1 Background

Compared to traditional manufacturing, rapid prototyping can be more efficient and less expensive in communicating product design. Prototypes can now be rapidly produced from 3D computer models. There are many different rapid prototyping technologies available and 3D laser scanning is commonly used for 3D digital model construction. 3D scanning is a process used for converting real world objects into digital models through the use of a sensor that captures the shapes and geometries of the objects [1]. The scanned image data can be processed in CAD software and saved as models for use in additive manufacturing. Most existing 3D imaging devices make use of 3D profiling cameras, sensors scanners, sonars or a combination thereof [2]. Figure 1A shows the essential components in a 3D scanning system. Many commercial scanners integrate each individual component into a one-piece device, as Figure 1B shows. Generally speaking, these sensors are high-cost and the processing speed is low because the scanning system acquires massive amounts of information from the workspace environment. A portable, inexpensive and quick processing 3D scanner can reduce time and cost in rapid prototyping.

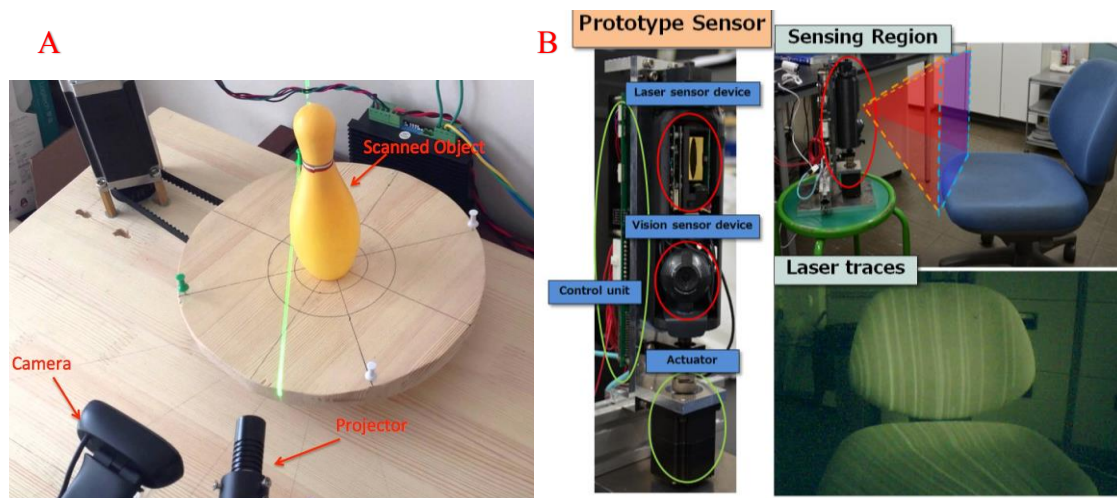


Figure 1: Components in 3D Scanning System (A) and Integrated 3D Scanner System (B) [3]

1.2 Working Concept of 3D Laser Scanners

There are several different approaches to 3D scanning. This section explains the most common method used by laser scanners. Scanners work with scanning programs or software. The sensor or camera acquires image data in the scanning process. Image data acquisition is a robust fitting procedure between the digital coordinate and the physical coordinate in the scan workspace [4]. In the 3D scanning system, the laser gives out laser light, and the laser light is reflected off the object, at a certain point or laser stripe. The sensor picks up the reflected laser light. The scanning program fits geometric relationships between the 3D coordinates of the laser stripe on the physical object and its digital coordinates in the image plan. Figure 2 shows the guide of how 3D laser scanner works.

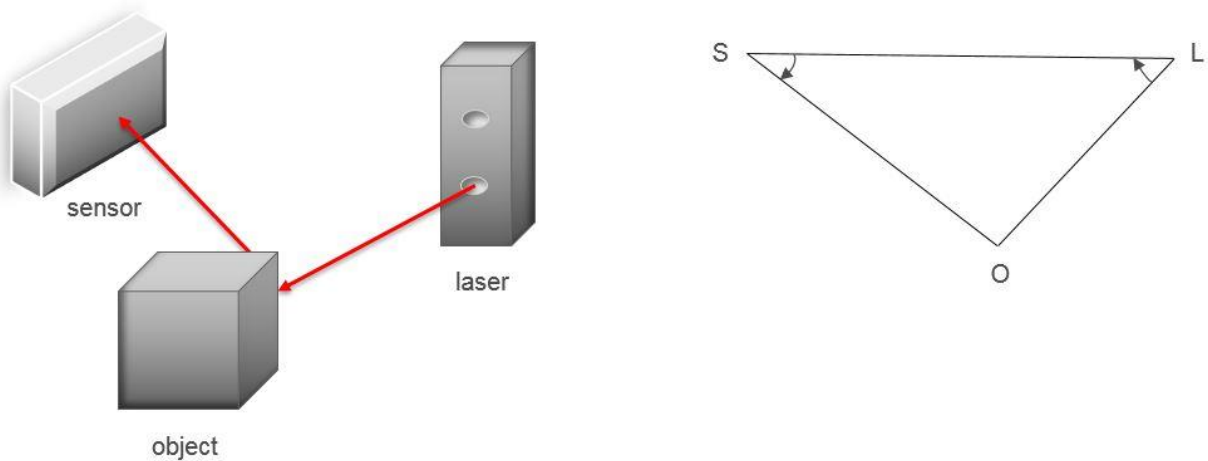


Figure 2: Sketch of 3D Laser Scanning System

There are several known parameters in this system. The positions of the sensor and the laser are known in x, y, z spatial coordinates. The angle of the emitted laser light is determined in the calibration process. Also, the system determines the angle at which the laser light is incident on the sensor. In this triangular system, there are two angles and one

length known. By trigonometric triangulation principle, the distance between the objects to the sensor can be captured by the scanning system, as the equation shown below:

$$\frac{\sin A}{a} = \frac{\sin B}{b} = \frac{\sin C}{c}$$

Once the distance between the point and the camera is determined, the position of this certain point can be calculated in a 3D coordinate system, and represented as (x, y, z). Scanning is a process of obtaining the positions of points on the surface as many as possible and form a point cloud, which is the image of the shape or geometries of the object. This is the most common method to obtain data by commercial 3D laser scanners in the market.

1.3 Purpose of the Research

The purpose of this research is to evaluate the technical performance capability of a DIY 3D laser scanner and compare it to the capability of a commercial scanner. The research is broken down into three phases: 1) to construct a portable, inexpensive 3D laser scanner. 2) Improve the accuracy of the scanner by optimizing the design. 3) Design a set of calibration objects to evaluate and compare the accuracy of the DIY 3D laser scanner.

1.4 Significance of the Research

3D laser scanners are widely used in industry. Honda is using 3D laser scanners ROMER Absolute Arm and PolyWorks software for quality inspection of plastic parts and Honeywell is also using 3D scanners for box dimension checking. The disadvantages of the commercial scanners are high cost and slow processing. Compared to the existing commercial scanners, this research will evaluate the performance of an inexpensive and portable 3D laser scanner with medium accuracy for daily rapid prototyping. The whole cost of the components to build up the DIY scanner is under 30 dollars. With the

manufacturing cost included, such as 3D printing, the overall budget is still under 150 dollars. The result of this study will provide a simple standard method for evaluating the accuracy of a 3D scanning system. In addition, the calibration object design will stand on its own as good innovation.

1.5 Outline of Thesis

This thesis describes the work completed to construct a 3D laser scanner, which is based on an original design published on Thingiverse, and the design of several calibration objects. The first chapter of the thesis provides background information about the existing 3D laser scanners and the working theory. The second chapter describes the design settings, components, open code sources and the scanning software used in the research. The third chapter discusses the design and the concept idea of the calibration objects. The fourth and final chapter provides future work on this topic. Appendices can be found at the end of the document.

Chapter 2: Design of the DIY 3D Laser Scanner

2.1 Overview of the DIY Laser Scanner

This research project is based on and developed from an existing original design on Thingiverse (by Ferretti) [5]. In this DIY scanning system, there are four line lasers used as projectors. Instead of using a sensor, a profiling camera Hercules HD Twist 5.0 MP 720P High-definition Mini Web Cam with 1280×720 resolution is used. The components in the scanner also include:

- Scanning arms (left and right)
- A camera holder (upper part and lower part)
- 4 laser holders
- Motor mount
- Four 80/20 profiles
- A corner
- A turntable
- A shaft
- A calibration part

Different from point lasers, the line laser light ensures that a laser trace is obtained every scanning step, instead of a point. The scanner does not need to change the camera height during the scanning process in order to get data on different surface layers as a result. The components in the scanner listed above are 3D printed with ABS material. The 3D printed parts are connected by 80/20 profiles. This enables the focus distance from the camera to the center of the table and the height of the turntable can be adjusted.

The turntable is laser cut with 1/8-inch thickness acrylic sheet. The overall design is about 50 cm long and wide, in 30 cm height. The initial setting of the laser scanner is shown

in the figure below. Some parts were modified as the research project developed and the redesign is illustrated in later chapters.

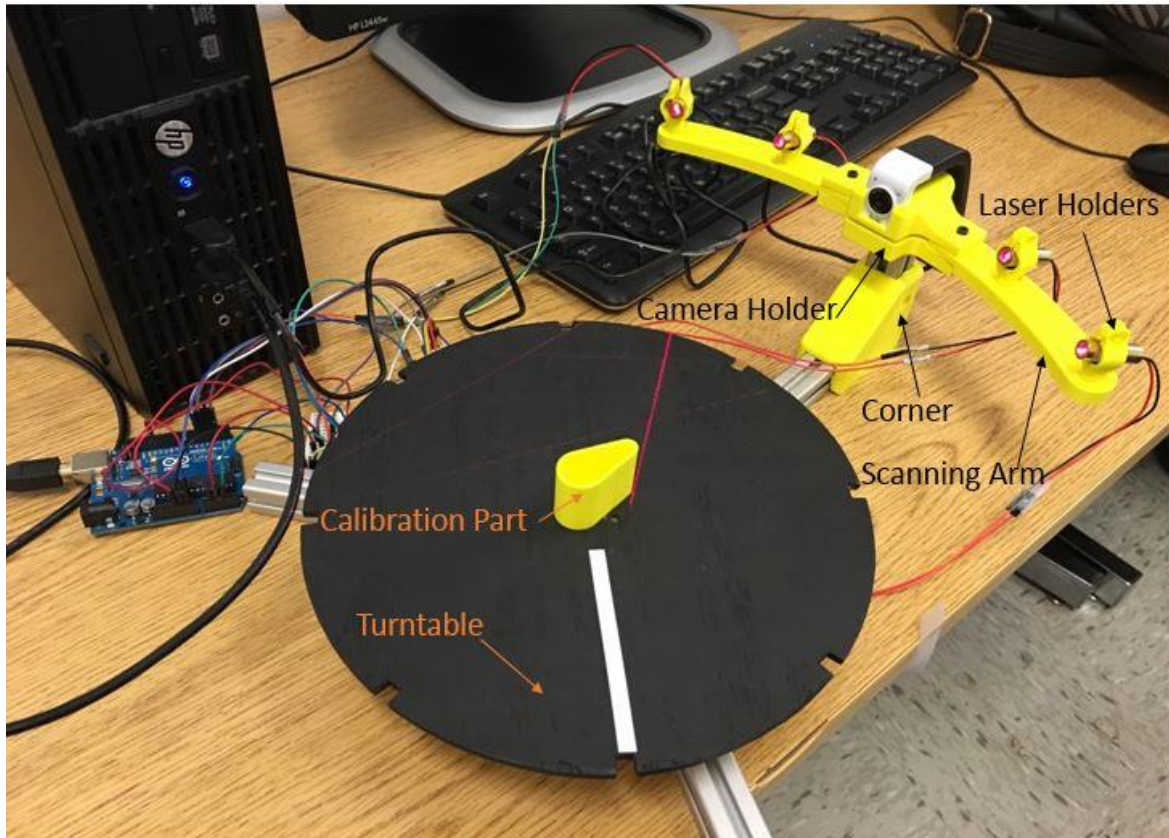


Figure 3: Top View of the DIY Scanner Physical Setting

2.2 Control System of the Design

An Arduino Nano v3.0 ATmega 328 board is used as a controller in the design. The hardware of the control system includes a stepper motor 28 BYJ-48 5VDC, a stepper motor controller ULN2003.

The Arduino sketch used to control the scanner is an open source on Thingiverse. It requires two libraries, AccelStepper_master and Adafruit_Motor_Shield to compile the main sketch. The two libraries are available on GitHub, and are also open source codes

programmed in C language. In addition, there is a configuration.h file that defines the names of the variables needed in the main program. The Arduino pins and electrical connections should be consistent with the variable names defined in this header file. The Arduino Sketch is attached in Appendix B.

A circuit is constructed and used to connect the lasers, motor and the controllers. A diagram is shown following:

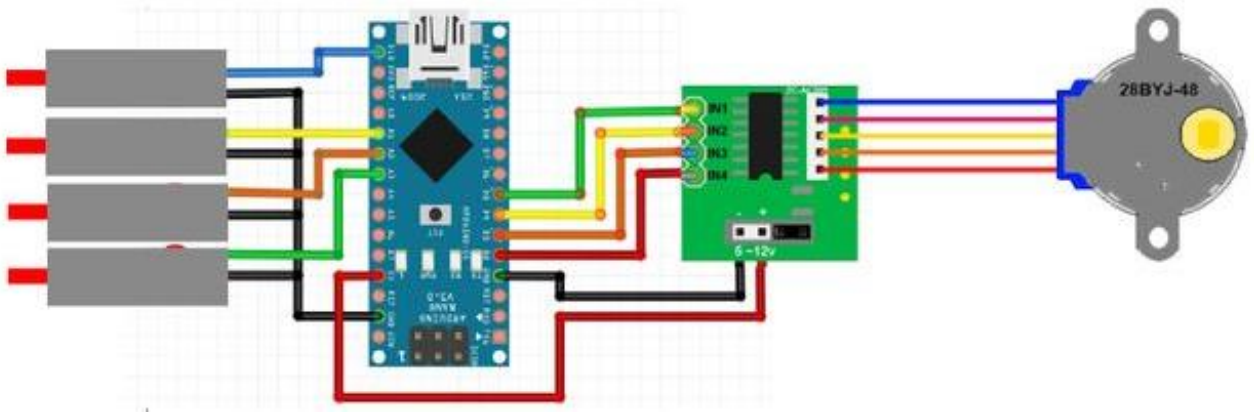


Figure 4: Circuit Sketch of the Control System [5]

The first laser is connected to digital pin 13 and the other three are connected to Analog pins from A1 to A3. The stepper motor is connected to digital pins from 2 to 5 as well as 0 and 5 V. The Arduino board in the hardware can be replaced with any standard Uno board. Also, the wiring can also be changed as long as the Arduino sketch is changed correspondingly to match the wiring in the configuration.h file.

According to the specification sheet of 28BYJ-48-5V stepper motor attached in Appendix D , the motor has 64 steps for one revolution and the stride angle is 5.625° . This means for one step, the turntable rotates 5.625° .

After the sketch is compiled, it can be tested and interacted with by the user through the use of the serial command window. The baud rate should be set to be 11520 bits and a line “Sardauscan v4.0” will pop up in the serial command window. Type “Sardauscan” in the window and the Arduino sketch should answer “yes”. Then test the stepper motor by typing “T R” with the number of steps desired. For example, if the user input “T R 100”, the motor will turn 100 steps. In addition, the four lasers are named as L0, L1, L2, and L4 in the Arduino sketch. “L 0 1” means turn the first laser on. Similarly, “L 0 0” will turn the first laser off. The same test can be applied to all four lasers. A pseudocode instruction about this test procedure is included in Appendix A and the details of the wirings are included in Appendix B and C.

2.3 Software and Interface

Just like commercial scanners in market, the DIY scanner also has a scanning software called “Sardauscan”, which is an open source software available on GitHub. The interface of Sardauscan is shown in the following figure:

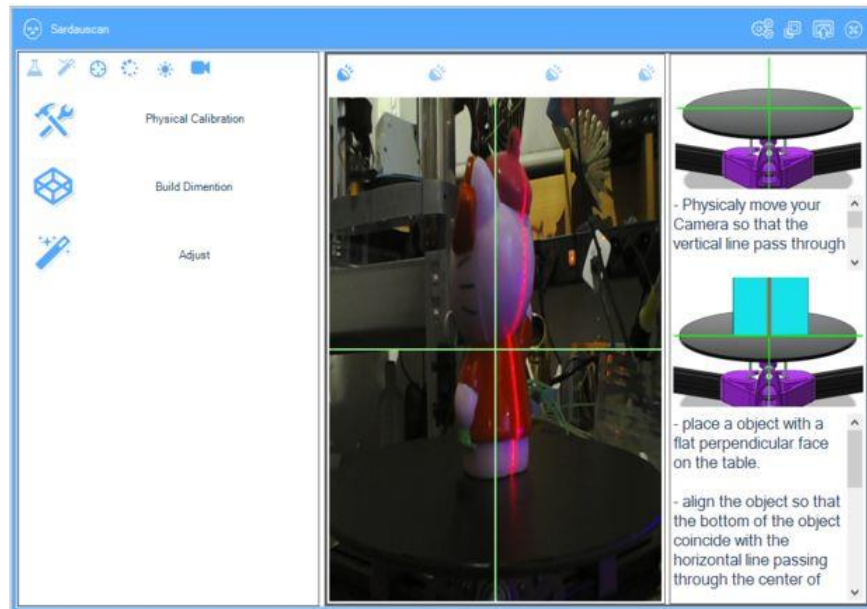


Figure 5: Main Interface of Sardauscan [5]

The Sardauscan scanning software can communicate with the Arduino sketch and through COM port connections. The camera is connected to Sardauscan scanning software through USB port. In the main interface, there are three red icons. They are used to connect the turntable, camera and four lasers to the software. After the connection, the scanner is ready to be calibrated. The calibration interface is as follows:

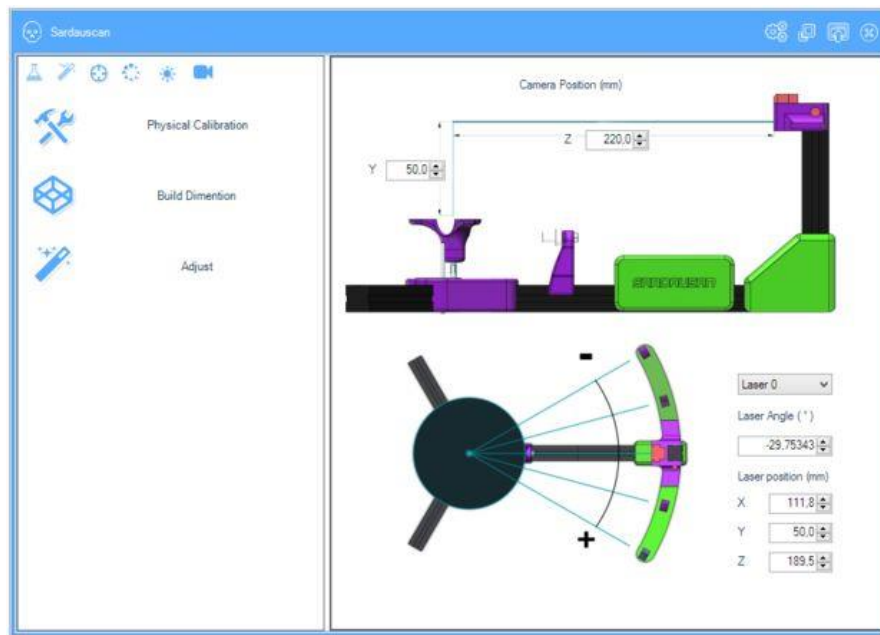


Figure 6: Manual Calibration Interface of Sardauscan [5]

The three icons listed on the left column on Figure 6 are:

- Physical Calibration
- Build Dimension
- Adjust Correction

The Calibration part is used in the calibration process. The camera is activated by clicking the “Physical Calibration” icon. There are two cursors (horizontal and vertical) across the preview window. Only the horizontal one can be moved in the window. The

position of the camera is adjusted so that the vertical cursor crosses the exact center of the table. The calibration part is then placed at the center of the table, and the horizontal cursor is aligned along the bottom of the alignment part. Right above the preview window there are four icons that activate each of the four lasers. Click on the icons and make sure the laser lines are aligned vertically on the flat face of the calibration part. After that, click on the build dimension icon and enter the dimensions as Figure 5 shows above. Now the scanner is ready for a quick scan. The icon is at the right top corner on the scan window.

The last correction step is after the acquisition of scan image. The image showed in the window is top view of the scanned object. Each of the four colors corresponds to the scan of a different laser. The goal of the adjustment is to align the four superimposed laser scans using the following process: select a laser in the combo box and rotate (left mouse button), scale (middle mouse button) and pan (right mouse button) the scan. The scan is used to adjust the four lasers in the window until they are aligned with each other as well as possible using a top view. After this correction step, the side view in the home main window should look like the actual physical geometry of the object scanned.

Chapter 3 Optimization and Redesign

The original design realized the basic function of the DIY scanner but still had some shortcomings in performance. The initial point cloud only showed a blurry shape of the object scanned. There are several potential sources of error. Misalignment of the four lasers in the calibration procedure would result in poor scan quality. Another possible factor is the quality of the mates between parts in the original design are not mated well. The camera holder was directly placed on the 80/20 aluminum profile without any connection. It could rotate during the scan, which can affect the data acquisition. In order to improve the scanning capability of the DIY scanner, the second phase of this research project is to improve the design for more accurate scanning.

3.1 Alignment Part

The original part used for calibration is a triangular part shown below:

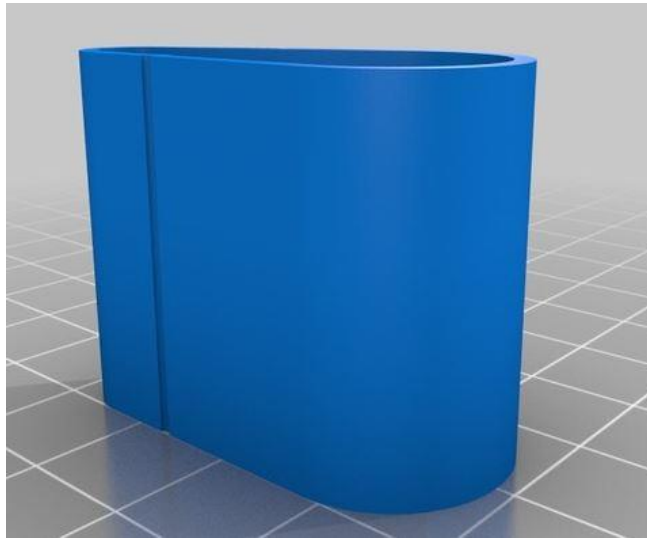


Figure 7: Original Calibration Part

The problem of this calibration part is that it is hard to align the four laser lights to get them aligned at the center of the table and perpendicular to the table. The alignment of the four laser lights can directly affect the scanning result since the camera seeks geometry information from the light traces. One way to improve the scanning capability of the DIY scanner is to replace the calibration part recommended to use in the original design on Instructables with an alignment part, which provides a reference for the four laser light traces to get aligned. The design selected for this purpose in 3D printed form is shown below:

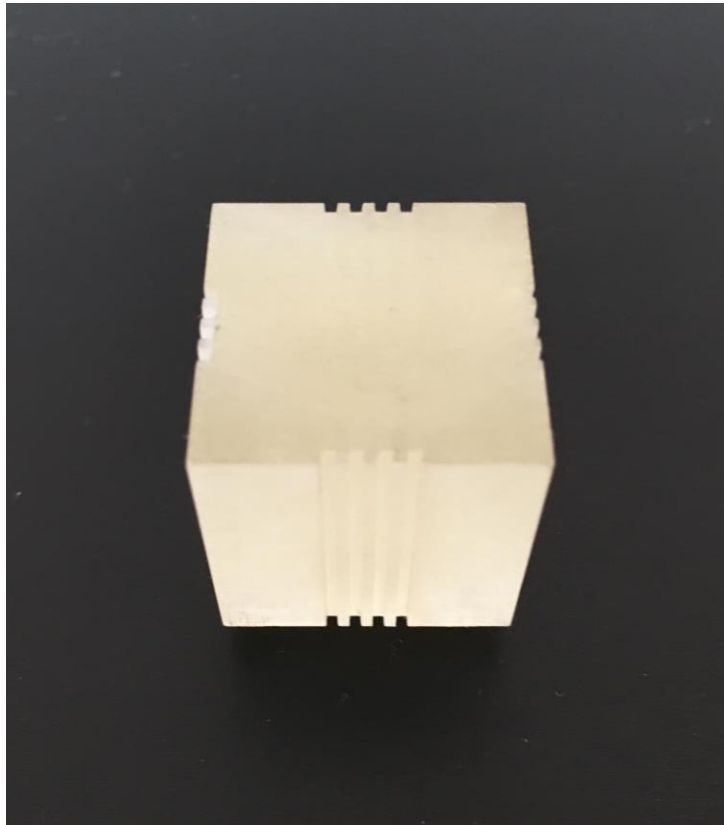


Figure 8: Alignment Part

On each side surface there are four grooves perpendicular to the table. The width of each groove is 1mm. During the scanner calibration process, every laser light is adjusted

and projected within the groove. This will help the laser lights get centered at one point. The grooves also provide a reference for adjusting the lasers perpendicular to the table.

3.2 Tolerance Analysis of the Center Position

The requirement to allow this alignment part to be used to align the lasers is that it sits at the center of the table. In order to satisfy this requirement, there is an assembly system of the shaft, alignment part, and the turntable. As figure 8 shows, there is a nub at the bottom of the alignment part, with a slot machined at the center. The turntable also has a slot with the same dimensions. The shaft goes through the turntable and gets assembled with the slot in the nub. The turntable has a soft mat on it and the nub is designed to assemble with the hole at the center of the mat, as shown in Figure 9.

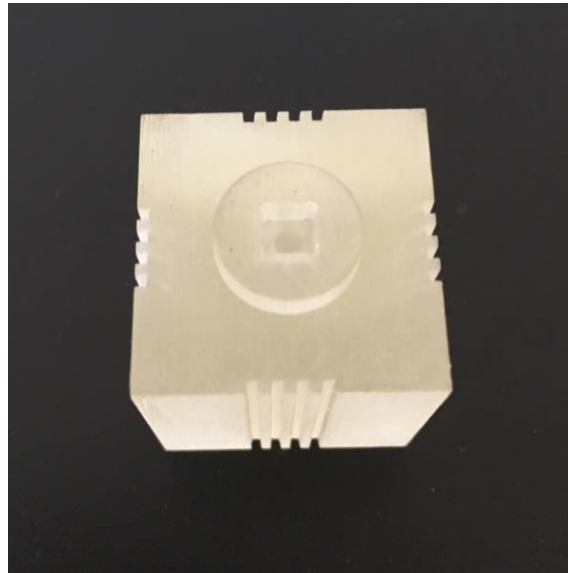


Figure 9: Alignment Part

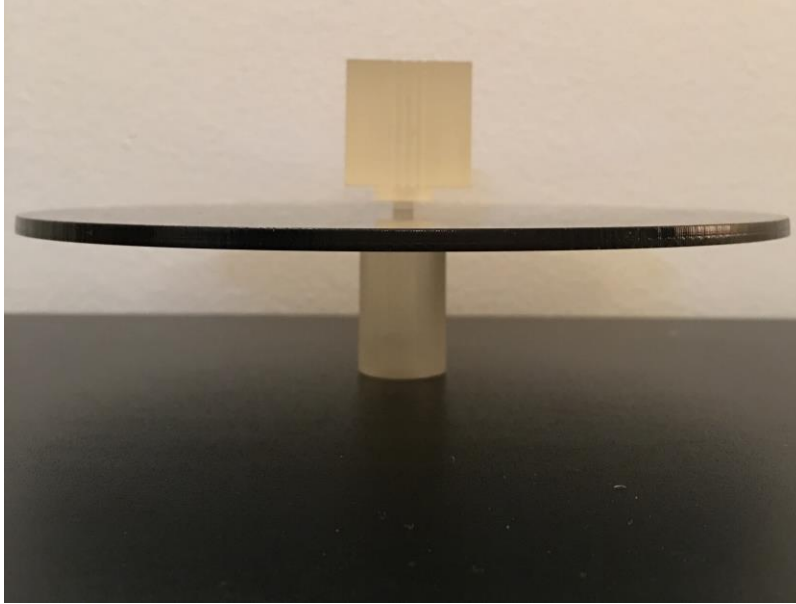


Figure 10: The Alignment Part and Turn Table Assembly

Machining and the resolution of the 3D printer introduce some variation from the desired position. The amount of variance determines how much deviation the alignment part can have from the center of the turntable. The alignment part is 3D printed using PolyJet Object30 Prime, with transparent photopolymer material RDG 720. The accuracy of this 3D printer in XY plan is 0.085 mm according to the specification sheet. The variance introduced by the machining operation of the slot is 0.001", with the length of the square. Assuming the centerline of the shaft to be a datum axis, the movement of the alignment part relative to the table depends on the tolerance calculated below:

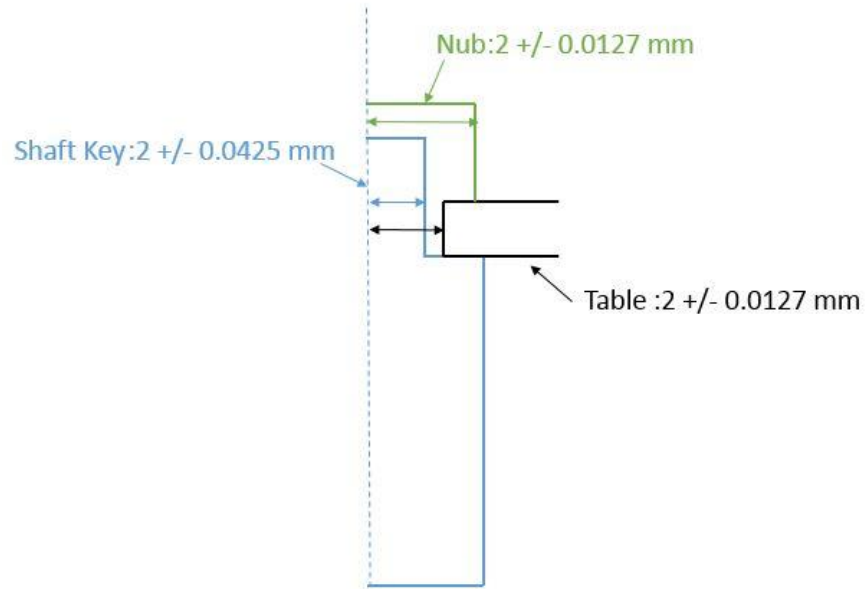


Figure 11: Scheme of the tolerance analysis

The upper and lower limits of the shaft key from the datum axis is:

$$\text{Upper limit} = \frac{0.085}{2} = 0.0425 \text{ mm}$$

$$\text{Lower limit} = -\frac{0.085}{2} = -0.0425 \text{ mm}$$

The upper and lower limits of the distance between the edge of the slot in the table and the datum axis:

$$0.001 \text{ inch} \times 25.4 \frac{\text{mm}}{\text{inch}} = 0.0254 \text{ mm}$$

$$\text{Upper limit} = \frac{0.0254}{2} = 0.0127 \text{ mm}$$

$$\text{Lower limit} = -\frac{0.0254}{2} = -0.0127 \text{ mm}$$

The dimensional variations of the assembly between the shaft and the table is:

$$\text{Upper limit} = 0.0425 + 0.0127 = 0.0552 \text{ mm}$$

Because this is a shaft-hole design, the lower limit of the assembly is 0.

Similarly, the dimensional variations of the assembly between the shaft and the nub is:

$$\text{Upper limit} = 0.0425 + 0.0127 = 0.0552 \text{ mm}$$

$$\text{Lower limit} = 0 \text{ mm}$$

Dimensional variation accumulation of the two assemblies above:

$$\text{Upper limit} = 0.0552 - 0 = 0.0552 \text{ mm}$$

$$\text{Lower limit} = 0 - 0.0552 = -0.0552 \text{ mm}$$

This variance means the alignment part can be off from the centerline of the shaft by maximum of 0.0552 mm.

3.3 Other Redesign of the Components

There are some other minor changes to the 3D printed parts of the DIY 3D laser scanner in this research project. The laser holders are downsized to mate better with the lasers. Also, there is a coupler designed to connect the camera holder and 80/20 profile. The pictures are shown in Figure 11 and 12.

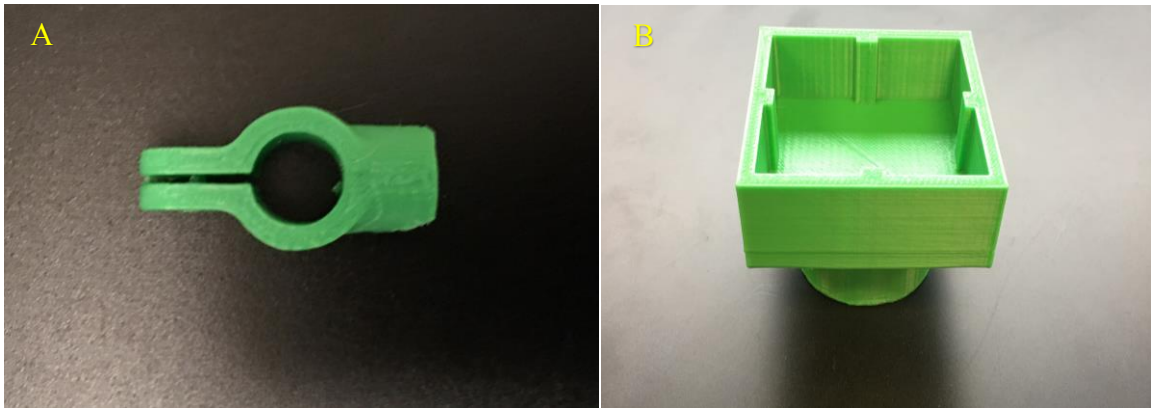


Figure 12: Downsized Laser Holders (A) and Coupler (B)

Chapter 4 Calibration Objects

4.1 Calibration Grid

A common method to determine the scanning workspace and calibrate the positional measurement accuracy is to use a calibration grid as shown in Figure 13. The calibration routine involves inputting distance in 3D space from the projector to every point on the grid into the scanning software. This is a tedious process. Additionally, another problem of the calibration grid is that it only evaluates the position of the specific points on the grid. There is a need for a standard or reference which can be used to determine the 3D location of points throughout the entire scanning workspace. A set of calibration objects can be used to help to implement this function.

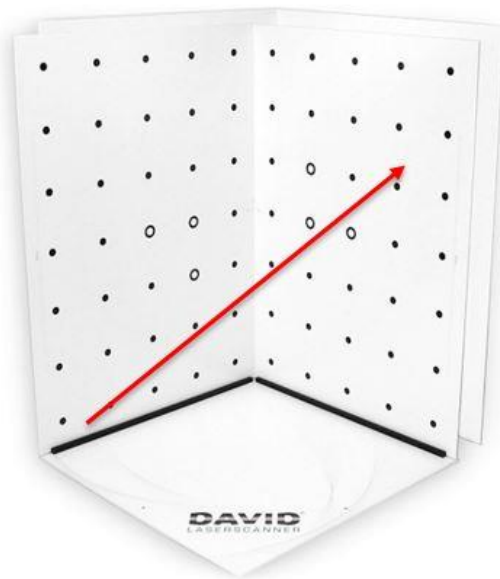


Figure 13: Commercial Calibration Grid [6]

4.2 Motion Capture

The design of calibration objects comes from motion capture and the concept is very similar. In motion capture system, the recorded motion capture data is mapped on a digital model in a 3D software, as shown in Figure 14. The motion capture workspace is calibrated by a wand with three marked points on it. The distance between any two marks is known. The wand is waved in the workspace and the coordinates of the points are defined and calibrated according to the distance between the marks on the wand.

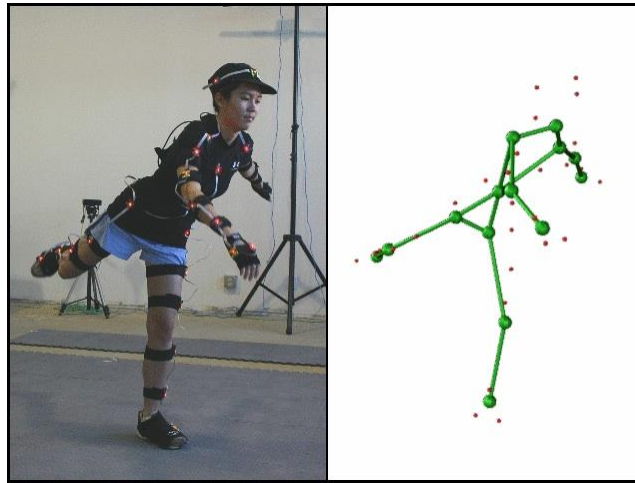


Figure 14: Motion Capture [7]



Figure 15: Calibration Wand [8]

4.3 Design of Calibration Objects

Similar to the calibration wand, the calibration objects also have specific known points or angle marks for comparison between the measured and true value of the features. The accuracy of the system can be represented as the deviations of the measurement data from the actual value. The functional objectives of the calibration objects are as the following:

- The features of the calibration objects are designed based on scanned objects in the real world.
- The calibration objects can be adapted to volume variations over a range of sizes.
- The geometry of the calibration objects should have both linear and non-linear contours.
- Both concave and convex geometries are included in the calibration objects.

In order to satisfy the requirements above, a set of two calibration objects were designed. Figure 16 show the initial designs:

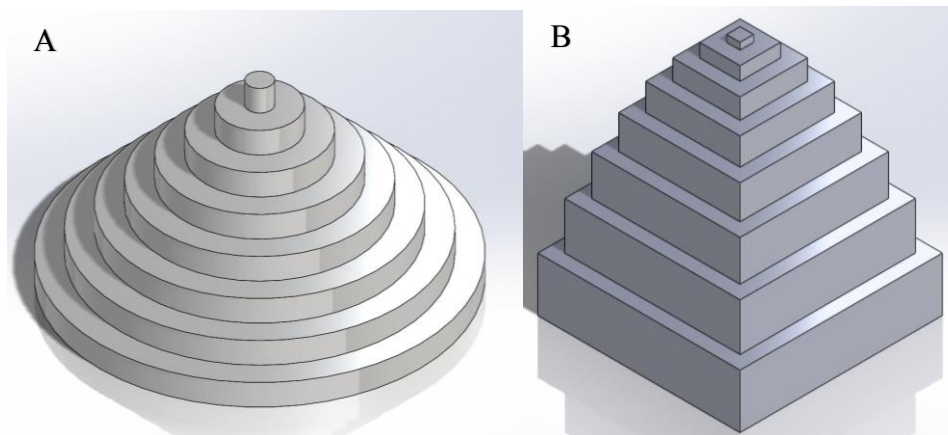


Figure 16: Conceptual Design of Calibration Object 1 (A) and Calibration Object 2 (B)

The steps on the calibration objects provide a linear increment in volume size. The measurement of each step height can be criteria to evaluate the scanning capability of a scanner. The two main shortcomings of the initial design are that object 1 cannot tell angular accuracy of a scan and there are no concave features in the design. The revised calibration objects are shown below:

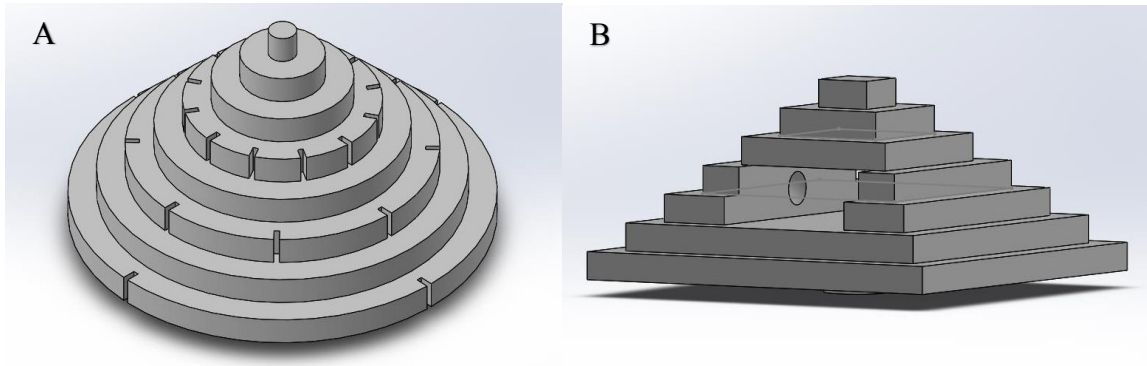


Figure 17: Final Design of Calibration Object 1 (A) and Calibration Object 2 (B)

The final design of calibration object 1 has thin slots at 30, 45 and 90 degree positions. This can help to test angular measurement accuracy. Also, calibration object 2 has a square cavity added and there are two semi spheres inside which can help to test how well the scanner can obtain information inside the concave geometry. The 3D printed calibration objects with transparent material are shown in the following figures:

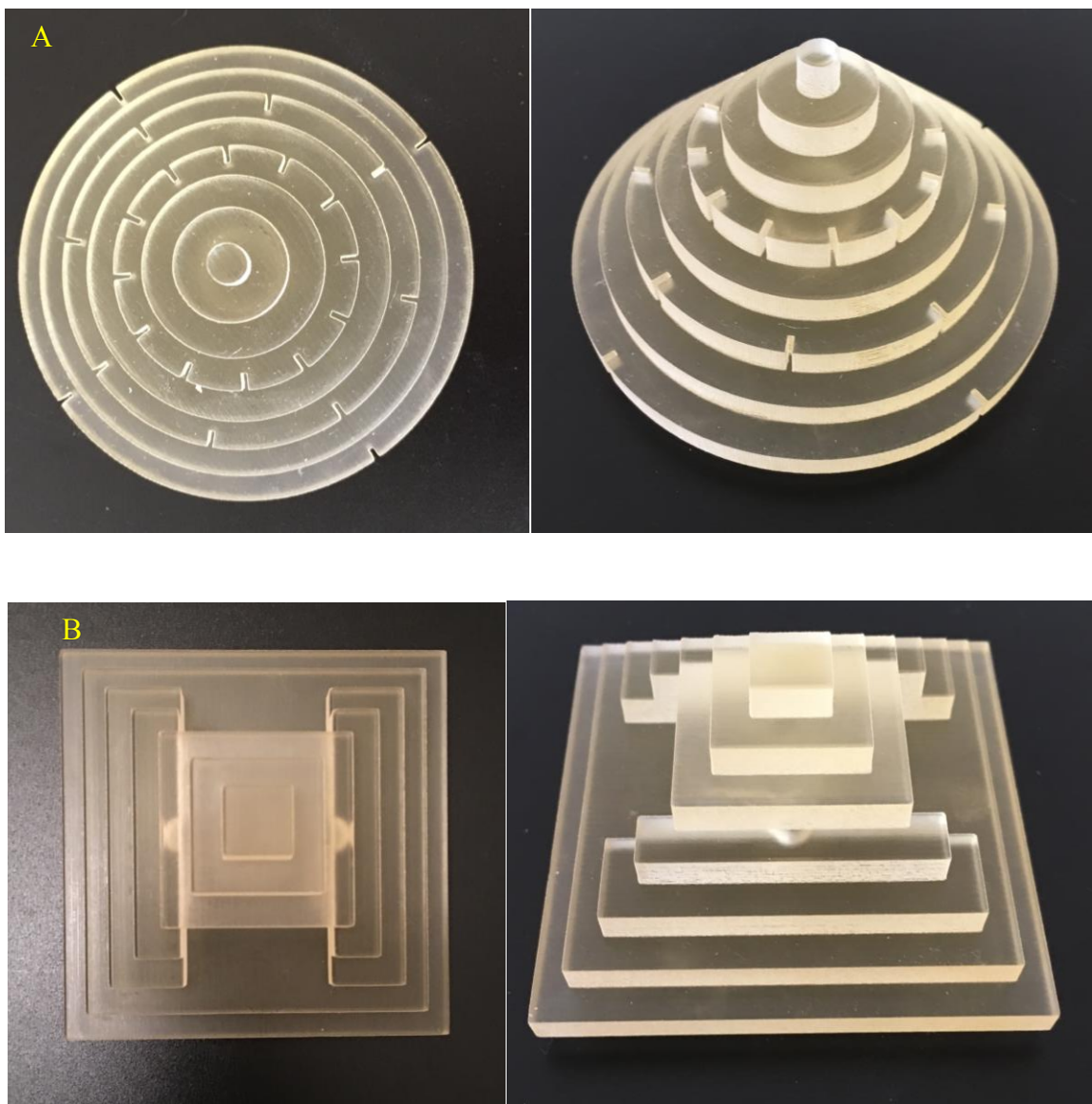


Figure 18: 3D printed Calibration Object 1 (A) and Calibration Object 2 (B)

Chapter 5 Result and Validation

The calibration objects are scanned by two commercial scanners, ROMER Absolute Arm and Structure Sensor. Then the two objects are scanned by the DIY laser scanner. The scan results of the two commercial scanners are then compared with the scan result of the DIY scanner. Some statistical analysis is done for the measurement using ROMER Absolute Arm, which gave the best results among the three scanners.

5.1 Scanning Capability of ROMER Absolute Arm

ROMER Absolute Arm is a 3D laser scanner working with PolyWorks software program. The point clouds of the two calibration objects scanned by the ROMER Absolute Arm and modeled in PolyWorks are shown in the following figures:

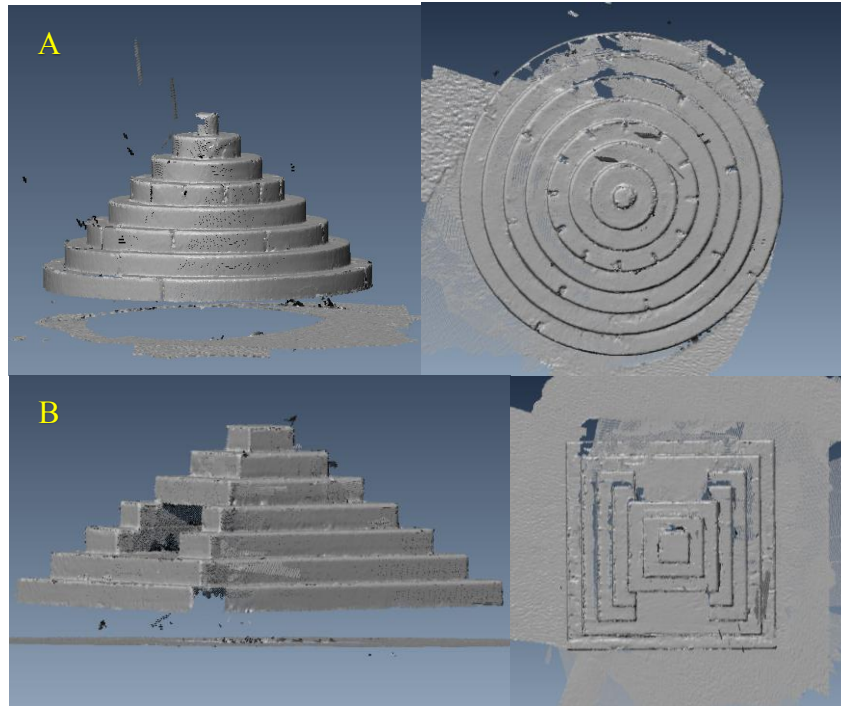


Figure 19: Point Clouds of Calibration Object 1 Top View and Side View (A)

Point Clouds Calibration Object 2 Top View and Side View (B)

The steps and slots are clearly visible to see in the point clouds above. In order to export the data as an image, the point cloud was converted to a mesh and saved as an IGES file, which can be opened in CAD software. The following figure shows the scan result in SolidWorks with the steps numbered for identification.

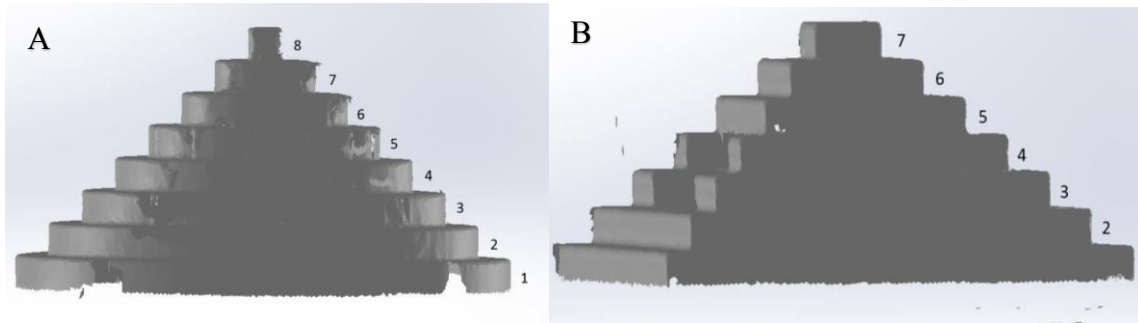


Figure 20: Scan Results of Calibration Object 1 (A) and 2 (B) in SolidWorks

The value of the individual local height of the same step slightly varies across the surface. For each step, three values of measurements were taken and the average value was calculated for the height of the step. The following table lists the measurements taken, compared with the actual physical size measured by a caliper:

Table 1: Measurement of Step Height of Calibration Object 1

Step #	1	2	3	4	5	6	7	8
Actual Physical Height (mm)	4.975	5.000	5.025	5.025	4.975	4.975	5.000	5.000
Measured Height (mm)	5.135	5.114	5.146	5.069	5.016	5.028	5.028	5.055
	5.078	5.109	5.036	4.996	5.145	5.017	5.057	5.058
	5.088	5.070	5.027	5.004	5.104	4.966	5.070	5.062
Average Measurement (mm)	5.100	5.098	5.070	5.023	5.088	5.004	5.052	5.058
Absolute Difference (mm)	0.125	0.098	0.045	0.002	0.113	0.029	0.052	0.058

For eight steps, the average absolute difference is:

$$\mu = \frac{0.125 + 0.098 + 0.045 + 0.002 + 0.113 + 0.029 + 0.052 + 0.058}{8} = 0.065 \text{ mm}$$

The standard deviation of the absolute difference is:

$$\delta = \sqrt{\frac{0.060^2 + 0.033^2 + 0.020^2 + 0.063^2 + 0.048^2 + 0.036^2 + 0.013^2 + 0.007^2}{7}}$$

$$= 0.043 \text{ mm}$$

Step 1, 3 and 5 have thin slots at 30, 45, and 90 degree positions. The angle between the slot positions are measured using a rotary table, as figure 21 shows. The values of measurement were taken in sexagesimal and converted to decimal degrees. Table 2 to 4 lists the measurements of the angle between the slot positions:

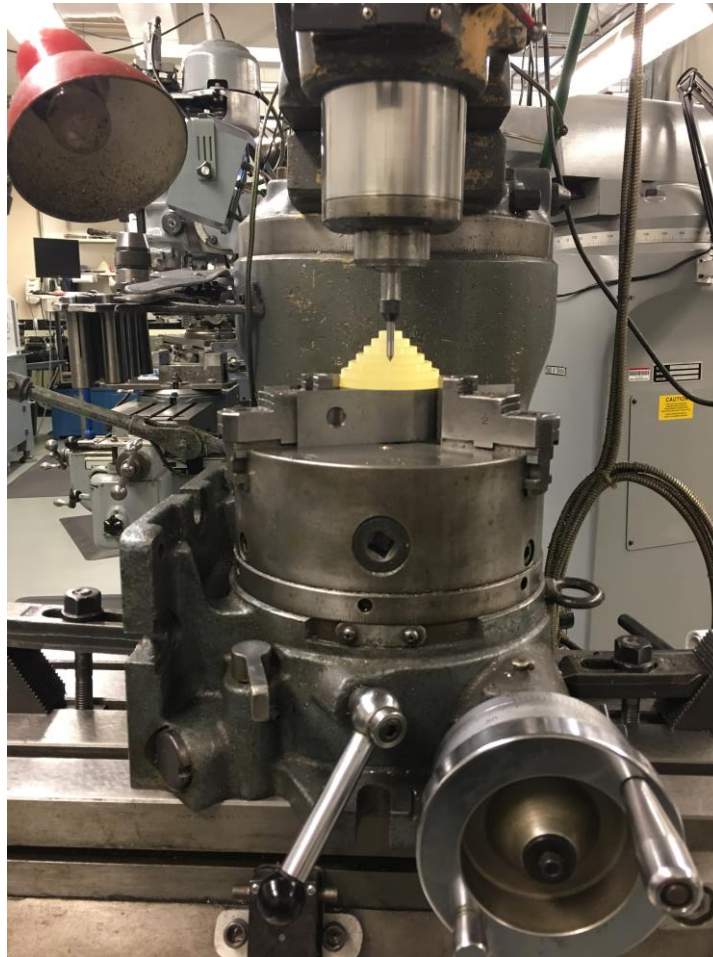


Figure 21: Setup of Physical Measurement of Angles on Rotary Table

Table 2: Measurement of Angular positions on Step 1 of Calibration Object 1

Step #	1			
Actual Physical angle (°)	90.41	90.12	90.42	90.25
Measured Angle (°)	90.42	90.24	90.28	90.16
	90.13	90.16	90.57	89.78
	90.04	90.22	90.10	90.20
Average Angle (°)	90.20	90.20	90.31	90.05
Absolute Difference (°)	0.21	0.08	0.11	0.2

The average absolute difference of the measurement is:

$$\mu = \frac{0.21 + 0.08 + 0.11 + 0.2}{4} = 0.15^\circ$$

The sample standard deviation of the absolute difference is:

$$\delta = \sqrt{\frac{0.06^2 + 0.07^2 + 0.04^2 + 0.05^2}{3}} = 0.06^\circ$$

Table 3: Measurement of Angular positions on Step 3 of Calibration Object 1

Step #	3							
Actual Physical angle (°)	45.29	44.94	45.36	45.18	45.10	44.93	45.45	45.04
Measured Angle (°)	45.68	45.20	45.38	45.23	45.01	44.79	45.66	45.14
	45.72	45.53	45.41	45.39	45.17	44.79	45.78	45.37
	45.24	45.15	45.30	45.32	45.16	44.92	45.54	44.96
Average Angle (°)	45.55	45.29	45.28	45.31	45.11	44.83	45.66	45.16
Absolute Difference (°)	0.26	0.35	0.08	0.13	0.01	0.10	0.21	0.12

The average absolute difference of the measurement is:

$$\mu = \frac{0.26 + 0.35 + 0.08 + 0.13 + 0.01 + 0.10 + 0.21 + 0.12}{8} = 0.16^\circ$$

The standard deviation of the absolute difference is:

$$\delta = \sqrt{\frac{0.10^2 + 0.19^2 + 0.08^2 + 0.03^2 + 0.15^2 + 0.06^2 + 0.05^2 + 0.04^2}{7}} = 0.11^\circ$$

Table 4: Measurement of Angular positions on Step 5 of Calibration Object 1

Step #	1	2	3	4	5	6	7	8	9	10	11	12
Actual Physical angle (°)	30.04	30.42	30.26	30.50	29.92	30.27	29.78	29.84	30.12	30.05	30.60	30.47
Measured Angle (°)	29.90	30.26	30.25	30.92	29.55	30.41	29.64	29.66	30.14	30.50	30.85	30.43
	30.14	30.25	30.02	30.71	29.43	30.22	29.54	30.04	30.41	30.71	30.73	30.31
	30.02	30.68	30.15	30.85	29.42	30.31	29.67	29.78	30.49	30.2	30.62	30.95
Average Angle (°)	30.02	30.40	30.14	30.83	29.47	30.31	29.62	29.82	30.35	30.47	30.73	30.56
Absolute Difference (°)	0.02	0.02	0.12	0.33	0.45	0.04	0.16	0.02	0.23	0.42	0.13	0.09

The average absolute difference μ of the measurement is:

$$\frac{0.02 + 0.02 + 0.12 + 0.33 + 0.45 + 0.04 + 0.16 + 0.02 + 0.23 + 0.42 + 0.13 + 0.09}{12}$$

$$= 0.17^\circ$$

The standard deviation of the absolute difference δ is:

$$\sqrt{\frac{0.15^2 + 0.15^2 + 0.05^2 + 0.16^2 + 0.28^2 + 0.13^2 + 0.01^2 + 0.15^2 + 0.06^2 + 0.25^2 + 0.04^2 + 0.08^2}{11}}$$

$$= 0.16^\circ$$

Similarly, same analysis is performed to the other calibration object. Calibration object 2 only has steps. The measured values of each step height and physical dimensions are listed in Table 5.

Table 5: Measurement of Step Height of Calibration Object 2

Step #	1	2	3	4	5	6	7
Actual Physical Height (mm)	5.025	5.075	5.025	5.025	5.000	5.000	5.025
Measured Height (mm)	4.905	4.951	4.987	4.922	5.022	5.014	5.130
	4.986	5.030	5.050	4.969	4.915	4.983	5.026
	4.935	5.021	4.985	4.912	4.927	5.039	4.985
Average Measurement (mm)	4.942	5.001	5.007	4.934	4.955	5.012	5.047
Absolute Difference (mm)	0.083	0.074	0.018	0.091	0.045	0.012	0.022

For seven steps, the average absolute difference is:

$$\mu = \frac{0.083 + 0.074 + 0.018 + 0.091 + 0.045 + 0.012 + 0.022}{7} = 0.049 \text{ mm}$$

The standard deviation of the absolute difference is:

$$\delta = \sqrt{\frac{0.034^2 + 0.025^2 + 0.031^2 + 0.042^2 + 0.004^2 + 0.037^2 + 0.027^2}{6}} = 0.033 \text{ mm}$$

The average of the difference between the measured value and physical actual value for each step or angle represents the variability of the scanning performance of ROMER Absolute Arm. The deviation is a good parameter to evaluate how reliable this variability result is. Seen from the result, ROMER Absolute Arm has better scanning capability in step dimensions than angular positions.

5.2 Scanning Capability of Structure Sensor

Another commercial 3D scanner that was used to scan the two calibration objects was the Structure Sensor.



Figure 22: The Structure Sensor [9]

The Structure Sensor is for use with an iPad and requires longer focal distance than ROMER Absolute Arm for a scan. In the data acquisition process, the scanning program used to work with the Structure Sensor was Scandy, an App available in Apple store. The scanning results are shown in figure 22.

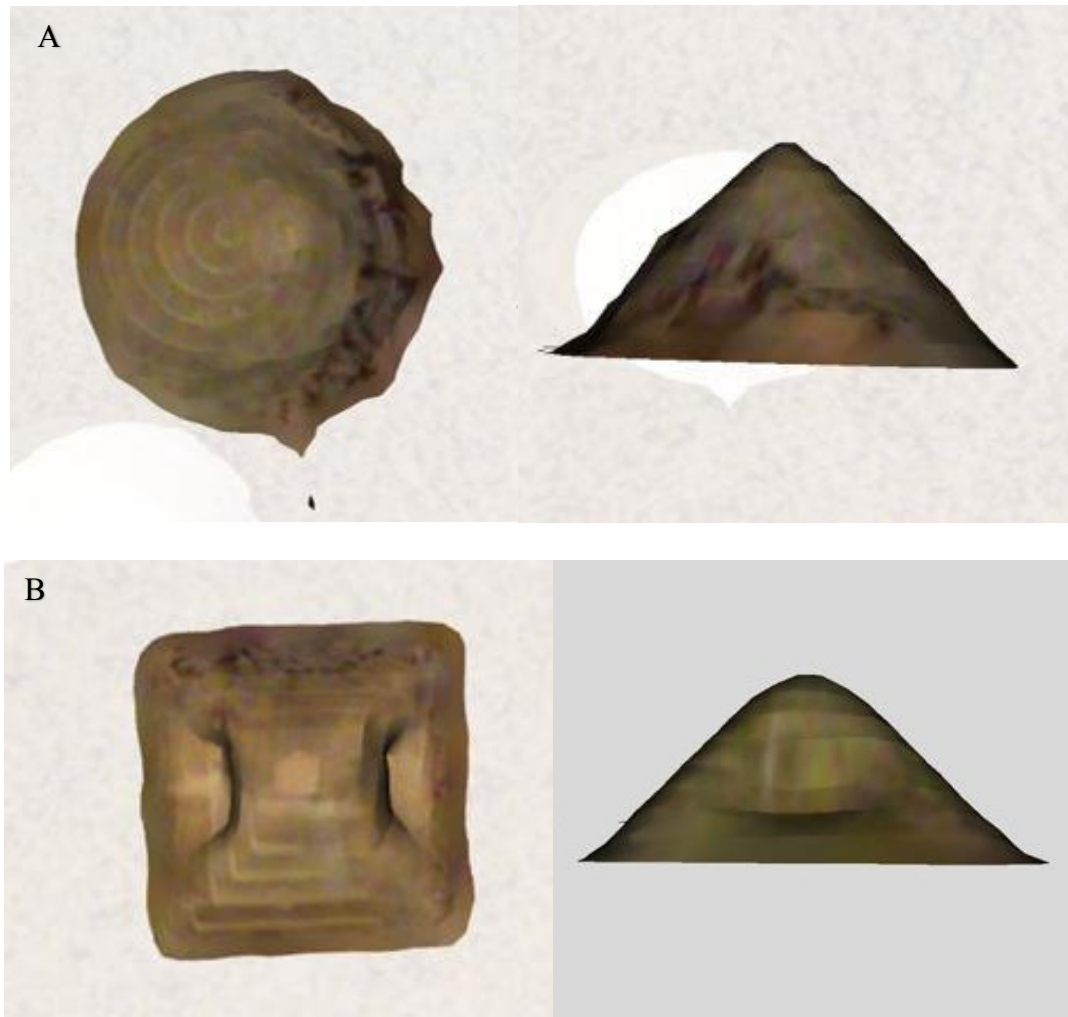


Figure 23: Scan of Calibration Object 1 Top View and Side View (A). Calibration Object 2 Top View and Side View (B).

Seen from the side view pictures above, the structure sensor was not able to capture the steps on the calibration objects. The top view pictures only show the blurry contours of each step. For calibration object 2, the structure sensor could only scan the rough concave geometry, instead of the whole open cavity. Although structure sensor is good at capturing human portrait and simulating real world physics, it did not perform well in scanning the details on calibration objects accurately. Compared to ROMER Absolute Arm, the structure sensor was not able to scan the small calibration objects.

5.3 Scanning Capability of the DIY 3D Laser Scanner

The DIY scanner has a better scan result than that of the Structure Sensor. The general geometries of the two calibration objects are clearly visible on Figure 23. Steps are clear though no slot was captured in the scan results. Compared to ROMER Absolute Arm, the DIY scanner was not able to obtain the same amount of details. An unclear shape of the square cavity on calibration object 2 was captured by the 3D DIY scanner. The laser light traces are not dense within the region of the square cavity, as shown in Figure 24.

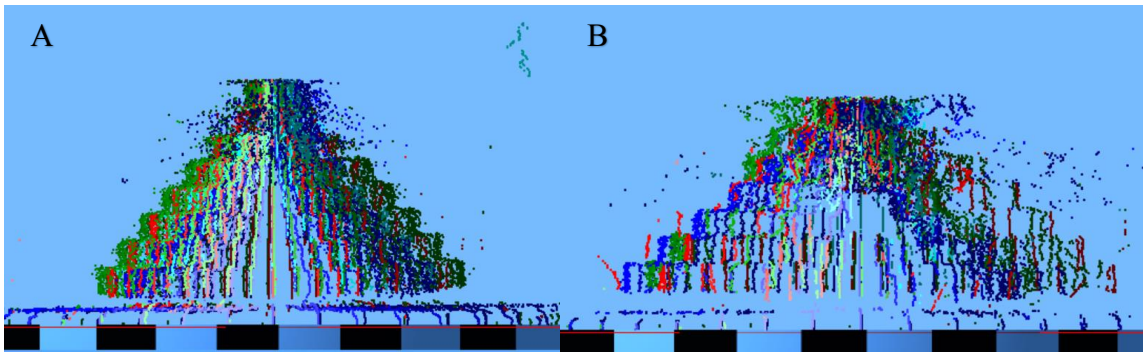


Figure 24: Scan of Calibration Object 1 Side View (A) and Calibration Object 2 Side View (B)

The scan result above has some noise, shown as scattered point clouds around the top of the calibration objects. This might be because the calibration objects are made of transparent material. Partial laser lights were refracted through the objects instead of being reflected to the camera. In order to validate this assumption, an aluminum box of similar size was scanned and compared with the result in Figure 23.

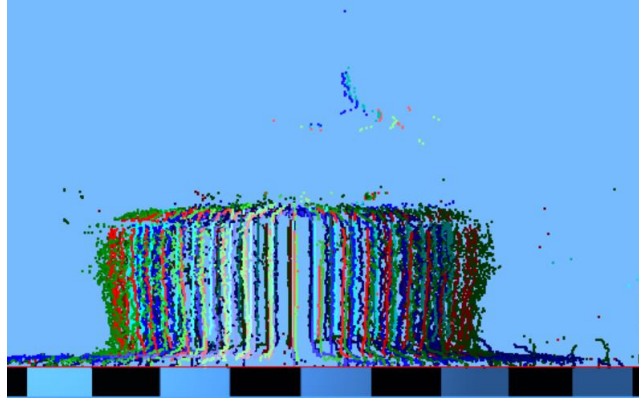


Figure 25: Scan of Aluminum box

The scan of this aluminum box was after the same calibration process as the scan of calibration objects. Apparently, there are fewer scattered points around the aluminum box, compared with the scan of calibration objects. This validates that transparent material increased the noise of the DIY laser scanner.

There are several potential errors that can affect the scan result of the DIY scanner. The alignment of lasers is the most crucial one. The perpendicularity of the lasers will affect the manual correction, which is the last step in calibration process. If any one of the lasers is off from the table center, the digital coordinate will not match well with the physical coordinate of the scan workspace. The scans of the four lasers will have knife edges and cannot align with each other well from the top view. Another serious problem with the scanner is vibration. The base of the scanner slightly vibrates when the turn table rotates. Mounts or lockers should be designed to make the scanner stable during scanning process.

Chapter 6 Conclusion and Future Work

The most important contribution of this study was that it developed a simple method to evaluate the scanning performance of scanners. It evaluates the calibration results performed by 2D grids in a reference to a set of 3D positions in the scan workspace.

Through the comparison between the scan results of the two commercial scanners, the calibration objects are able to differentiate the scanning capabilities. The Structure Sensor requires long focal distance for scans, which adapts better to large objects or human profiles. Although the structure sensor is good for daily use in capturing dense geometries in real-time, it did not give good scan results using the calibration objects. The DIY 3D laser scanner performed better than the Structure Sensor in capturing the steps. ROMER Absolute Arm captured most feature details on the calibration objects among the three scanners, with variable focal distance. It can be concluded that ROMER Absolute Arm has better accuracy with the test of the calibration objects.

On the other hand, there are some design modifications that could improve the performance of the calibration objects. The two calibration objects are 3D printed with transparent photopolymer material RDG 720 in this study. One potential issue with this material is that the transparent material affects the scan result with laser scanners. Partial laser lights will go through the clear material and have refraction. The calibration objects can be made using other manufacturing methods such as CNC machining, which would eliminate this problem.

Another modification needed is for the alignment part. The equally distributed grooves can help to get laser lights aligned at a certain point but this point can still be off

from the table center. A cylinder shape alignment part will be better to correlate with the angles between the positions of the lasers and the camera.

Future work could include additional design improvements to the scanner, improving laser alignment and eliminating vibration problems. Also, calibration objects design could be made simpler for manufacturing and easier to get physically measured. The geometry should also be adjusted according to different manufacturing methods. In addition, more features such as spheres, cavities, gaps or bridges can be added to calibration objects to test how well the scanner can obtain information over small objects or inside concave geometries. The two calibration objects in this study are a good start and there can be more designs developed based on them.

References

- [1] "3D Scanners", [online], Available: <http://www.rapidform.com/3d-scanners/>.

- [2] R. Macknoja, A. Chavez-Aragon, P. Payeur, R. Laganier, "Calibration of a Network of Kinect Sensors for Robotic Inspection over a Large Workspace", *Robot Vision (WORV), 2013 IEEE Workshop on*, pp. 184-190, 2013.

- [3] [Image. "Capturing the Aviator with 3D Scanning"](#), [online] available: <https://www.google.com/search>.

- [4] V. Niola, C. Rossi, S. Savino, S. Strano, "A method for the calibration of a 3-D laser scanner", *SinceDirect*, 27 (2), 479-484.

- [5] "Sub 30\$ 3D laser scanner", [online], Available: <http://www.thingiverse.com>.

- [6] R. Macknoja, A. Chavez-Aragon, P. Payeur, R. Laganier, "Calibration of a Network of Kinect Sensors for Robotic Inspection over a Large Workspace", *Robot Vision (WORV), 2013 IEEE Workshop on*, pp. 184-190, 2013.

- [7] Adam G. Kirk, James F. O'Brien, and David A. Forsyth. "Skeletal Parameter Estimation from Optical Motion Capture Data". In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR) 2005*, pages 782-788, June 2005.

- [8] Calibration Wands, [online], available: <http://www.optitrack.com/products/tools/>

[9] The Structure Sensor, [online], available: <http://structure.io/getstarted>

Appendices

Appendix A: Pseudocode Instruction to Test the Arduino Sketch

Appendix B: Arduino Sketch to Control the DIY 3D Laser Scanner

Appendix C: Configuration Header File with Variable Name Defined

Appendix D: Specification Sheet of Motor 28BYJ-48

Appendix A

Pseudocode Instruction to Test the Arduino Sketch

Commands :

identity

sardauscan =>ask if this is a sardaucan => response "yes" //

laser :

L => ask laser configuration => response number of lasers

L x => ask status of laser index X

=> 'x' laser number

L x y => turn laser on or off

'x' laser number

'y' state: 0=off 1=on

Table :

T => table position

T S => return the number of steps for a revolution

T C => Set Current position as 0 (for absolute positioning

T R xxx => rotate relatively xxx steps

T A xxx => turn position absolute xxx steps

unknow command

=> response "Hun?"

*/

Appendix B

Arduino Sketch to Control the DIY 3D Laser Scanner

```
#include "SerialCommand.h"

#include <AccelStepper.h>

#include "configuration.h"

AccelStepper stepper1(HALFSTEP, motorPin1, motorPin3, motorPin2, motorPin4);

SerialCommand SCmd;

int RotationToSteps(int rotation)
{
    return rotation*STEP_BY_MINMOVE;
}

int StepsToRotation(int rot)
{
    return rot/STEP_BY_MINMOVE;
}

void Identification()
{
    Serial.println("yes");
}

void Hun()
{
    Serial.println("Hun?");
}
```



```

void TableCommand()
{
    char *arg;

    arg = SCmd.next();

    if (arg != NULL)
    {
        if(arg[0]=='R' || arg[0]=='r') // T R
        {
            char *arg2 = SCmd.next();

            int pos=atoi(arg2);

            stepper1.move(RotationToSteps(pos));

            stepper1.runToPosition();

            Serial.print("RELATIVE ROTATION :");

            Serial.print(pos);

            Serial.print(" => ");

            Serial.println(StepsToRotation(stepper1.currentPosition()));
        }

        else if(arg[0]=='A' || arg[0]=='a') // T A
        {
            char *arg2 = SCmd.next();

            int pos=atoi(arg2);

            stepper1.moveTo(RotationToSteps(pos));

            stepper1.runToPosition();

            Serial.print("ABSOLUTE ROTATION ");

```

```

        Serial.print(pos);

        Serial.print(" => ");

        Serial.println(StepsToRotation(stepper1.currentPosition()));
    }

    else if(arg[0]=='S' || arg[0]=='s') //T S
    {

        Serial.print("REVOLUTION STEPS ");

        Serial.print(" => ");

        Serial.println(StepsToRotation(REVOLUTION_STEP));
    }

    else if(arg[0]=='C' || arg[0]=='c') //T C
    {

        stepper1.setCurrentPosition(0);

        Serial.print("RESET CURRENT POSITION ");

        Serial.print(" => ");

        Serial.println(StepsToRotation(stepper1.currentPosition()));
    }

    else
    {

        Serial.print("Unknown Table command :");

        Serial.println(arg);
    }
}

else {

```

```

    Serial.print("Position ");

    Serial.println(StepsToRotation(stepper1.currentPosition()));
}
}

int getLaserPin(int laserIndex)
{
    if(laserIndex==0)
        return LASER_PIN_1;
    else if(laserIndex==1)
        return LASER_PIN_2;
    else if(laserIndex==2)
        return LASER_PIN_3;
    else if(laserIndex==3)
        return LASER_PIN_4;
    else
        return (-1);
}

void LaserCommand()
{
    char *arg;
    arg = SCmd.next();
    if (arg != NULL)
    {

```

```

int laserIndex=atoi(arg);

char *arg2 = SCmd.next();

if (arg2 == NULL)
{
    Serial.print("LASER_STATE: ");

    Serial.print(laserIndex);

    int pin =getLaserPin(laserIndex);

    Serial.print("(");

    Serial.print(pin);

    Serial.print(") = ");

    if(pin>=0)

        Serial.println(digitalRead(pin));

    else

        Serial.println(-1);
}
else
{
    int laserState = atoi(arg2);

    int pin =getLaserPin(laserIndex);

    if(pin>=0)

        digitalWrite( pin, laserState==1?HIGH:LOW);

    Serial.print("SET_LASER: ");

    Serial.print(laserIndex);

    Serial.print("(");

```

```

    Serial.print(pin);

    Serial.print(" = ");

    Serial.println(digitalRead(pin));

}

}

else

{

    Serial.print("LASER_COUNT: ");

    Serial.println(LASER_COUNT);

}

}

```

```

void setup() {

    stepper1.setMaxSpeed(500.0);

    stepper1.setAcceleration(200.0);

    stepper1.setSpeed(400);

    stepper1.moveTo(0);

    stepper1.runToPosition();

    Serial.begin(SERIAL_BAUD);


    pinMode(LASER_PIN_1, OUTPUT);

    pinMode(LASER_PIN_2, OUTPUT);

    pinMode(LASER_PIN_3, OUTPUT);

    pinMode(LASER_PIN_4, OUTPUT);
}

```

```

digitalWrite(LASER_PIN_1, LOW);

digitalWrite(LASER_PIN_2, LOW);

digitalWrite(LASER_PIN_3, LOW);

digitalWrite(LASER_PIN_4, LOW);

SCmd.addCommand("sardauscan",Identification);

SCmd.addCommand("T",TableCommand);

SCmd.addCommand("t",TableCommand);

SCmd.addCommand("L",LaserCommand);

SCmd.addCommand("l",LaserCommand);

SCmd.addDefaultHandler(Hun);

Serial.println(FIRMWARE_VERSION);

Serial.flush();

}

void loop() {

    SCmd.readSerial();  // We don't do much, just process serial commands

    Serial.flush();

}

```

Appendix C

Configuration Header File with Variable Name Defined

```
#ifndef CONFIGURATION_h

#define CONFIGURATION_h

#define FIRMWARE_VERSION "Sardauscan V0.1a"

#define SERIAL_BAUD 115200

//57600

// Motor definitions

#define motorPin1 2 // IN1 on the ULN2003 driver 1

#define motorPin2 3 // IN2 on the ULN2003 driver 1

#define motorPin3 4 // IN3 on the ULN2003 driver 1

#define motorPin4 5 // IN4 on the ULN2003 driver 1

//tips (from Mark Benson)

//If anyone else is having problems with a BYJ48 stepper not doing anything,

//change the HALFSTEP value to 4 & REVOLUTION_STEP to 2048

#define HALFSTEP 4

#define REVOLUTION_STEP 2048

#define STEP_BY_MINMOVE 4 // move by STEP_BY_MINMOVE (to avoid power loss when
position is between step)

#define LASER_COUNT 4

#define LASER_PIN_1 A1 //yellow
```

```
#define LASER_PIN_2 A2 //orange
```

```
#define LASER_PIN_3 A3 //green
```

```
#define LASER_PIN_4 13 //blue
```

```
#endif
```


Appendix D

Specification Sheet of Motor 28BYJ-48

28BYJ-48 – 5V Stepper Motor

The 28BYJ-48 is a small stepper motor suitable for a large range of applications.



Rated voltage :	5VDC
Number of Phase	4
Speed Variation Ratio	1/64
Stride Angle	5.625°/64
Frequency	100Hz
DC resistance	50Ω±7%(25°C)
Idle In-traction Frequency	> 600Hz
Idle Out-traction Frequency	> 1000Hz
In-traction Torque	>34.3mN.m(120Hz)
Self-positioning Torque	>34.3mN.m
Friction torque	600-1200 gf.cm
Pull in torque	300 gf.cm
Insulated resistance	>10MΩ(500V)
Insulated electricity power	600VAC/1mA/1s
Insulation grade	A
Rise in Temperature	<40K(120Hz)
Noise	<35dB(120Hz, No load, 10cm)
Model	28BYJ-48 – 5V

